# MSBD 5008 Homework 2

Pranav A, 20478966

# 1 Problem 1

## 1.1 Initial Adoption

The nodes $\{f, e, h, i\}$ will switch behaviours. Table 1.1 indicates the steps at the time those nodes will switch.

| Step | Switching Node | Threshold |
|---|---|---|
| 1 | f | 0.4 |
| 2 | e | 0.5 |
| 3 | h | 0.5 |
| 3 | i | 0.67 |

Table 1: Steps of nodes switching and their current threshold at that time

## 1.2 Cluster obstacle

The cluster $\{g, j, k\}$ has a cluster density greater than $\frac{3}{5}$ which is stopping the cascade.

## 1.3 Solving cluster obstacle

Clearly the edge between the cluster and initial adopters is required. This will reduce the cluster density and allows the cascade to happen. Thus there could be edge between $\{c - g\}$ or $\{d - j\}$

# 2 Problem 2

Basically we have to select the nodes from each cluster which has a density greater than $q$ and which have the most number of connections in that cluster.

## 2.1 Nodes which are capable of causing a cascade

The nodes $\{6, 7, 11\}$ can cause a cascade.

## 2.2 More nodes which are capable of causing a cascade

The nodes $\{4, 9, 16\}$ can cause a cascade.

## 2.3 Clusters in network

The clusters with density greater than $\frac{1}{2}$ are:

1. $\{1, 4, 5, 6, 10\}$
2. $\{3, 7, 8, 9, 12\}$
3. $\{11, 13, 14, 15, 16\}$

## 2.4 Number of initial adopters

The network has three disjoint clusters with density of $\frac{1}{2}$. Also, the switching threshold is also $\frac{1}{2}$.

In order to cause a complete cascade, the cluster densities in the remaining network should be less than $1 - q$.

This shows that the network needs **at least three initial adopters** (one in each cluster) in order to disrupt the obstacle and allow the cascade to happen.

If we choose only 2 initial adopters, there will be at least one cluster in remaining network which has a density of $1 - q$, which will not cause a complete cascade.

# 3 Problem 3

The figure 1 describes the centrality calculations which are obtained from the code attached below. Additional operations like making dataframes, slicing lists etc. are omitted in the code for the clarity.

| | Degree Centrality | Betweeness Centrality | Closeness centrality | Eigenvector centrality |
|---|---|---|---|---|
| **0** | 0.484848 | 0.437635 | 0.568966 | 0.355483 |
| **16** | 0.060606 | 0.000000 | 0.284483 | 0.023635 |

Figure 1: The nodes and the corresponding centralities calculations in Zachary Karate Network

```python
import networkx as nx

# Graph from Networkx
G=nx.karate_club_graph()

# Calculating normalized degree centrality
nx_deg_centrality = nx.degree_centrality(G)

# Calculating betweenness centrality
nx_betwn_centrality = nx.betweenness_centrality(G)

# Calculating closeness centrality
nx_close_centrality = nx.closeness_centrality(G)

# Calculating eigenvector centrality
nx_eigen_centrality = nx.eigenvector_centrality(G)
```

# Notes

1. This given solution of the assignment follows the HKUST honour code. Although assignment has been discussed with other peers, the solutions are my own.

2. Kindly give feedback on this assignment on how to write up the solutions more elegantly.